

Amendments of the Specification

Please replace Paragraph [0011] with the following amended Paragraph [0011]:

[0011] In known programmable logic, four-to-one multiplexers ($N = 4$) are created by chaining together two four-input look-up tables in the manner described in more detail below. For $N > 4$, using binary encoding, trees of four-to-one and/or two-to-one multiplexers are created. The most efficient case for a fully-populated one-level tree (i.e., four four-to-one multiplexers feeding a single four-to-one multiplexer) involves five multiplexers, or ten look-up tables, to implement a sixteen-to-one multiplexer. This works out to $L = 0.625N$. The situation may be somewhat better where the tree is not fully populated. Thus, for $N = 5$, one four-to-one multiplexer and one two-to-one multiplexer, or three look-up tables, are required. This works out to $L = 0.6N$. Even for $N = 6$ or $N = 7$, two four-to-one multiplexers (four four-input look-up tables) can be used. This works out to $L \approx 0.67N$ for $N = 6$ or $L \approx 0.57N$ for $N = 7$. For $N = 8$, two four-to-one multiplexers and one two-to-one multiplexer, or five four-input look-up tables, can be used. This works out to $L = 0.625N$, as in the case of $N = 16$. Similar numbers obtain for $9 \leq N \leq 15$.